Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) A computer implemented method of automatically generating Electronic Data Interchange (EDI) documents by a trading partner comprising the steps of:

receiving, by the trading partner, a standard data model comprising EDI related data for a plurality of transactions;

generating from the standard data model, by the trading partner, data definitions for a self-describing markup language corresponding to each transaction of the EDI related data;

generating self-describing markup language data using a data definition from the generated data definitions for the self-describing markup language corresponding to an EDI transaction and corresponding application data related to EDI; and

automatically generating, by the trading partner, an EDI document based on the self-describing markup language data,

wherein the step of generating data definitions further comprises, for each transaction, generating data definitions for the self-describing markup language, a separate data model to read in data, a separate data model to read out data, and a map component file.

2. (Currently Amended) The method according to claim 1, wherein the step of generating data definitions comprises receiving user input of an EDI standard, a version of the standard, a transaction set, and mapping rules for the standard data model.

3. (Original) The method according to claim 1, wherein the step of generating data definitions comprises receiving a user input of an EDI standard.

4. (Original) The method according to claim 1, wherein the step of generating data definitions comprises receiving a user input of a version of a standard.

5. (Original) The method according to claim 1, wherein the step of generating data definitions comprises receiving a user input of a transaction set.

6-9. (Cancelled)

10. (Original) The method according to claim 1, wherein the generated EDI document conforms to an ANSI X12 standard.

11. (Original) The method according to claim 1, wherein the generated EDI document conforms to an UN EDIFACT standard.

12. (Original) The method according to claim 1, wherein the self-describing markup language comprises eXtensible MarkUp Language (XML).

13. (Previously Presented) A system for automatically generating Electronic Data Interchange (EDI) documents by a trading partner, the system comprising:

a standard data model comprising EDI related data of a plurality of transactions;

a computer implemented first generator that generates, from the standard data model, data definitions for a self-describing markup language corresponding to each transaction of the EDI related data;

a computer implemented second generator that generates self-describing markup language data using a data definition for the self-describing markup language corresponding to an EDI transaction and corresponding application data related to the EDI; and

a computer implemented translator that automatically generates an EDI document based on the self-describing mark up language data,

wherein the first generator generates, for each of the plurality of transactions, data definitions for the self-describing markup language, a separate data model to read in data, a separate data model to read out data, and a map component file.

14. (Original) The system according to claim 13, wherein the self-describing markup language comprises XML and wherein the first generator is a Data Type Definition Generator (DTD Generator).

15-18. (Cancelled)

19. (Original) The system according to claim 13, wherein the first generator further comprises a user interface for user input of an EDI standard, a version of the standard, and a transaction set prior to generating the EDI document.

20. (Original) The system according to claim 13, wherein the first generator further comprises a user interface for user input of an EDI standard prior to generating the EDI document.

21. (Original) The system according to claim 13, wherein the first generator further comprises a user interface for user input of a version of the standard prior to generating the EDI document.

22. (Original) The system according to claim 13, wherein the first generator further comprises a user interface for user input a transaction set prior to generating the EDI document.

23. (Original) The system according to claim 13, wherein the generated EDI document conforms to an ANSI X12 standard.

24. (Original) The system according to claim 13, wherein the generated EDI document conforms to an UN EDIFACT standard.

25. (Original) The system according to claim 13, wherein the self-describing markup language comprises eXtensible MarkUp Language (XML).

26. (Cancelled)

27. (Previously Presented) Program code on a computer readable medium, that is executable by a computer for generating Electronic Data Interchange (EDI) documents by a trading partner, the program code configured to cause the computer to perform the following steps:

receiving, by the trading partner, a standard data model comprising EDI related data for a plurality of transactions;

generating from the standard data model, by the trading partner, data definitions for a self-describing markup language corresponding to each transaction of the EDI related data;

generating self-describing markup language data using a data definition from the generated data definitions for the self-describing markup language corresponding to an EDI transaction and corresponding application data related to EDI; and

automatically generating, by the trading partner, an EDI document based on the self-describing markup language data,

wherein the step of generating data definitions further comprises, for each transaction, generating data definitions for the self-describing markup language, a separate data model to read in data, a separate data model to read out data, and a map component file.

28. (Previously Presented) The program code according to claim 27, wherein the self-describing markup language comprises XML and wherein the first generator comprises a Data Type Definition Generator (DTD Generator).

29-32. (Cancelled)

33. (Currently Amended) The program code according to claim 27, wherein the step of generating data definitions comprises receiving user input of an EDI standard, a version of the standard, ~~and~~ a transaction, and mapping rules for the standard data model.

34. (Previously Presented) The program code according to claim 27, wherein the step of generating data definitions comprises receiving user input of an EDI standard.

35. (Previously Presented) The program code according to claim 34, wherein the step of generating data definitions comprises receiving user input of the version of the EDI standard.

36. (Previously Presented) The program code according to claim 27, wherein the generated EDI document conforms to the ANSI X12 standard.

37. (Previously Presented) The program code according to claim 27, wherein the generated EDI document conforms to the UN EDIFACT standard.

38. (Previously Presented) The program code according to claim 27, wherein the self-describing markup language comprises eXtensible MarkUp Language (XML).

39. (Previously Presented) A computer implemented method of automatically generating Electronic Data Interchange (EDI) documents, by a trading partner, comprising the steps of:

receiving, by the trading partner, a standard data model containing EDI related data for a plurality of transactions;

receiving a manual entry of parameters related to an EDI document format;

generating from the standard data model and the manual entry of parameters, by the trading partner, data definitions for the self-describing markup language corresponding to each transaction of the EDI related data and the received manually entered parameters;

generating self-describing markup language data using the data definition for the self-describing markup language corresponding to an EDI transaction and corresponding application data related to EDI; and

automatically generating, by the trading partner, an EDI document based on the self-describing markup language data,

wherein the step of generating data definitions further comprises, for each transaction, generating data definitions for the self-describing markup language, a separate data model for read in data, a separate data model for read out data, and a map component file.

40-43 (Cancelled).

44. (Currently Amended) The method according to claim 39, wherein the step of receiving a manual entry of parameters comprises receiving user input of an EDI standard, a version of the standard, a transaction set, ~~and~~ a direction, and mapping rules for the standard data model.

45. (Original) The method according to claim 39, wherein the step of receiving a manual entry of parameters comprises receiving user input of an EDI standard.

46. (Original) The method according to claim 39, wherein the step of receiving a manual entry of parameters comprises receiving user input of a version of the EDI standard.

47. (Original) The method according to claim 39, wherein the step of receiving a manual entry of parameters comprises receiving user input of a transaction set.

48. (Original) The method according to claim 39, further comprising one data type definition for each transaction of each EDI standard used when generating EDI documents.

49. (Previously Presented) A computer implemented method of automatically generating data in a self-describing markup language format from received EDI data, comprising the steps of:

receiving EDI data from a component;

retrieving a self-describing markup language data definition corresponding to a transaction type of received EDI data; and

automatically generating self-describing markup language data based on the received EDI data and the self-describing markup language data definition,

prior to said receiving step, generating data definitions corresponding to each transaction type from a standard data model of EDI related data,

wherein the generating data definitions step comprises, for each transaction, a data definition for the self-describing mark up language, a separate EDI data model to read in data, a separate self-describing mark up language data model to read-out data, and a map component file.

50. (Cancelled)

51. (Currently Amended) The method according to claim 49, further comprising, prior to said retrieving step, receiving user input of an EDI standard, a version of the standard, and a transaction set in generating the self-describing markup language data definition, and mapping rules for the standard data model.

52. (Original) The method according to claim 49, further comprising, prior to said retrieving step, receiving a user input of an EDI standard in generating the self-describing markup language data definition.

53. (Original) The method according to claim 52, further comprising, prior to said retrieving step, receiving a user input of a version of the EDI standard in generating the self-describing markup language data definition.

54. (Original) The method according to claim 49, further comprising, prior to said retrieving step, receiving a user input of a transaction set in generating the self-describing markup language data definition.

55. (Original) The method according to claim 49, wherein the received EDI data conforms to the ANSI X12 standard.

56. (Original) The method according to claim 49, wherein the received EDI data conforms to the UN EDIFACT standard.

57. (Original) The method according to claim 49, wherein the generated self-describing markup language comprises eXtensible MarkUp Language (XML).

58. (Currently Amended) A ~~system~~ computer program product for automatically generating data in a self-describing markup language format from received EDI data, the ~~system~~ computer program product embodied in computer readable media executable by a computer, the computer program product comprising:

a component for transmitting EDI data;

a translator that receives a self-describing markup language data definition corresponding to a transaction type of received EDI data;

wherein the translator that automatically generates the self-describing markup language data based on the received EDI data and the self-describing markup language data definitions,

wherein the ~~receiver~~ translator receives the self-describing markup language data definition generated by a generator from a standard data model comprising EDI related data for a plurality of transactions, the data definition comprising, for each transaction, a data definition for the self-describing mark up language, a separate EDI data model to read in data, a separate self-describing mark up language data model to read-out data, and a map component file.

59. (Cancelled)

60. (Currently Amended) The ~~system~~ computer program product according to claim 58, wherein the generator further comprises a user interface for user input of an EDI standard, a version of the standard, and a transaction set prior to generating the self-describing markup language format.

61. (Currently Amended) The ~~system~~ computer program product according to claim 58, wherein the generator further comprises a user interface for user input of an EDI standard prior to generating the self-describing markup language format.

62. (Currently Amended) The ~~system~~ computer program product according to claim 61, wherein the generator further comprises a user interface for user input of a version of the EDI standard prior to generating the self-describing markup language format.

63. (Currently Amended) The ~~system~~ computer program product according to claim 58, wherein the generator further comprises a user interface for user input of a transaction set prior to generating the self-describing markup language format.

64. (Previously Presented) A program code on a computer readable medium that is executable by a computer for automatically generating data in a self-describing markup language data from received EDI data, the program code configured to cause the computer to perform the following steps:

receiving EDI data from a component;

retrieving a self-describing markup language data definition corresponding to a transaction type of received EDI data; and

automatically generating self-describing markup language data based on the received EDI data and the self-describing markup language data definition,

prior to said receiving step, generating data definitions corresponding to each transaction type from a standard data model comprising EDI related data for a plurality of transactions,

wherein the generating data definitions step comprises, for each transaction, a data definition for the self-describing mark up language, a separate EDI data model to read in data, a separate self-describing mark up language data model to read-out data, and a map component file.

65. (Cancelled)

66. (Currently Amended) The program code according to claim 64, further comprising, prior to said retrieving step, receiving user input of an EDI standard, a version of the standard, ~~and~~ a transaction set in generating the self-describing markup language data definition, and mapping rules for the standard data model.

67. (Previously Presented) The program code according to claim 64, further comprising, prior to said retrieving step, receiving user input of an EDI standard in generating the self-describing markup language data definition.

68. (Previously Presented) The program code according to claim 67, further comprising, prior to said retrieving step, receiving user input of the version of the EDI standard in generating the self-describing markup language data definition.

69. (Previously Presented) The program code according to claim 64, further comprising, prior to said retrieving step, user input of a transaction set in generating the self-describing markup language data definition.

70. (Previously Presented) The program code according to claim 64, wherein the received EDI data conforms to the ANSI X12 standard.

71. (Previously Presented) The program code according to claim 64, wherein the received EDI data conforms to the UN EDIFACT standard.

72. (Previously Presented) The program code according to claim 64, wherein the generated self-describing markup language comprises eXtensible MarkUp Language (XML).

73. (Cancelled).